Fig. 1

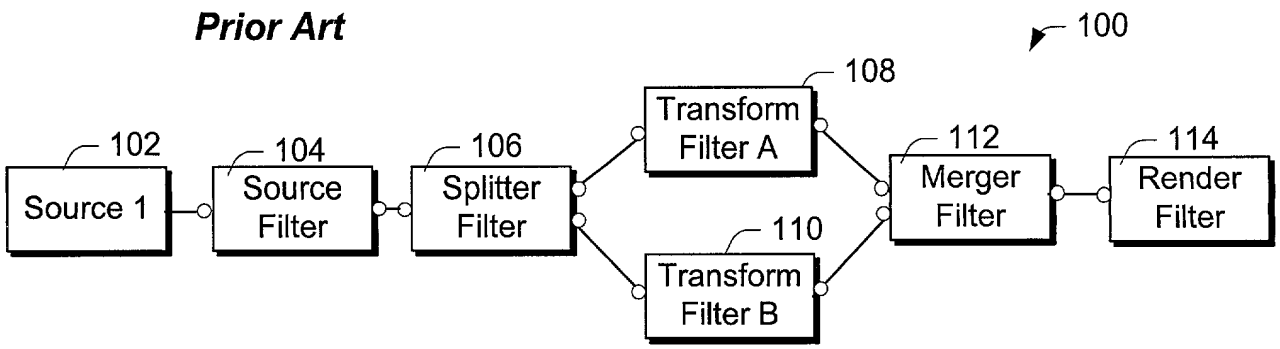
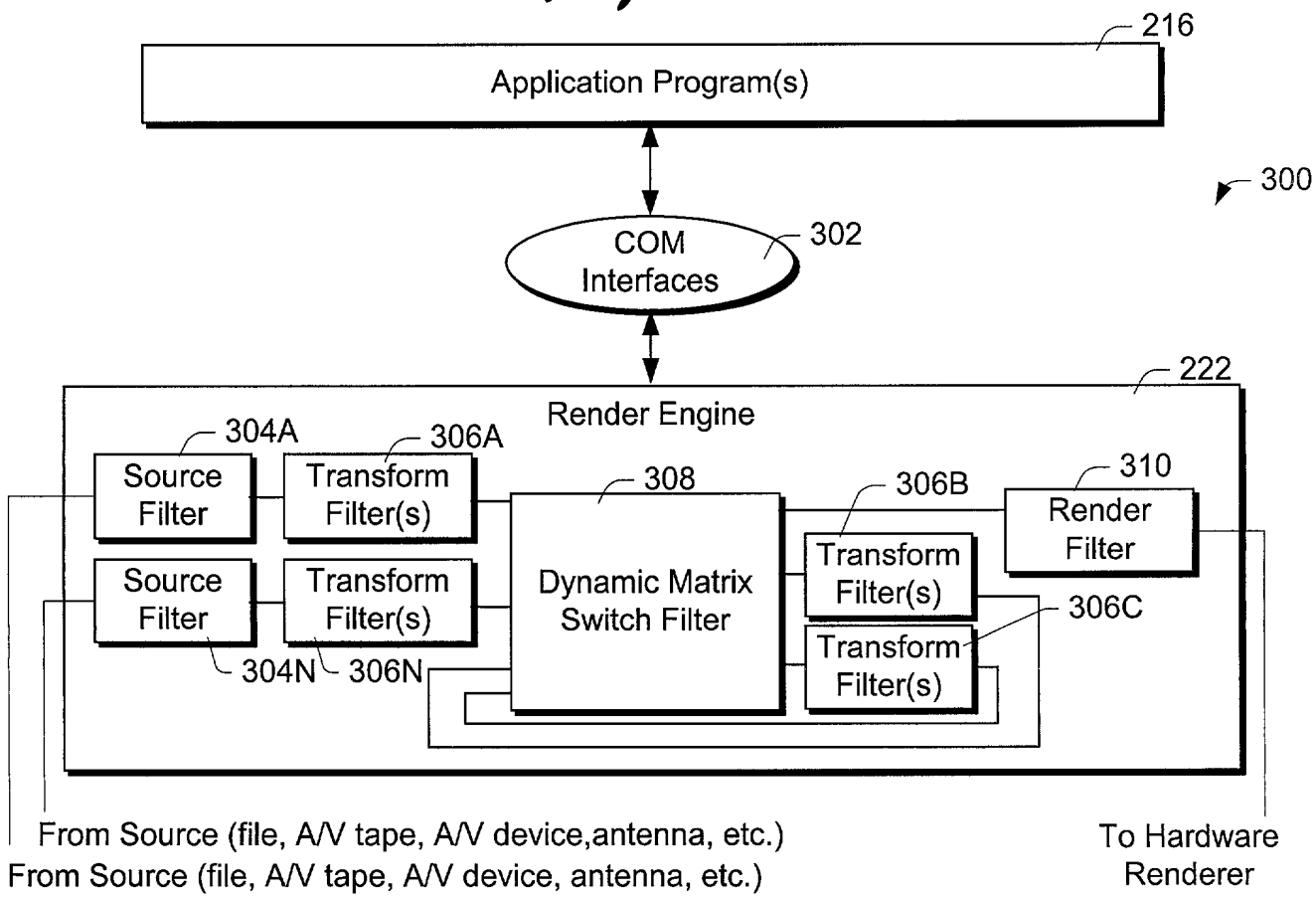

Prior Art
100
102
Source 1
104
Source Filter
106
Splitter Filter
108
Transform Filter A
110
Transform Filter B
112
Merger Filter
114
Render Filter

Fig. 3


216
Application Program(s)
300
COM Interfaces
302
222
Render Engine
304A
Source Filter
306A
Transform Filter(s)
Source Filter
Transform Filter(s)
304N
306N
308
Dynamic Matrix Switch Filter
306B
Transform Filter(s)
Transform Filter(s)
306C
310
Render Filter
From Source (file, A/V tape, A/V device,antenna, etc.)
From Source (file, A/V tape, A/V device, antenna, etc.)
To Hardware Renderer

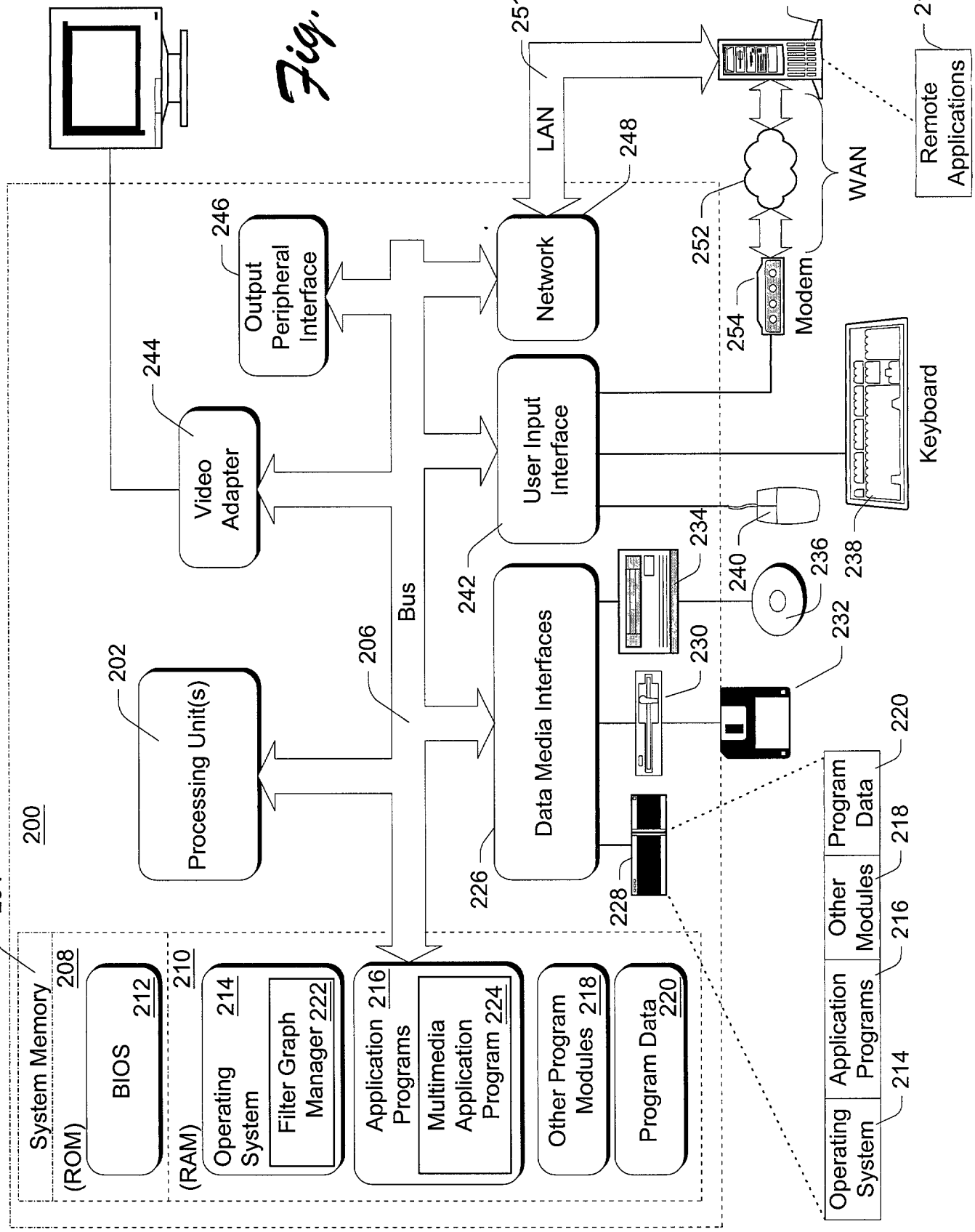
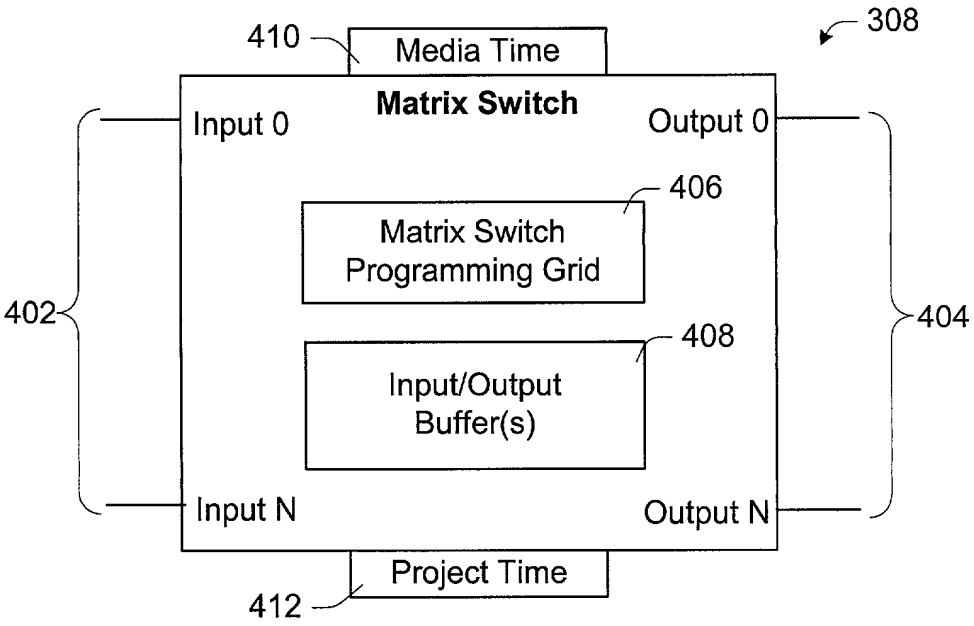

200
204
System Memory
(ROM)
208
BIOS
212
(RAM)
210
Operating System
214
Filter Graph Manager 222
Application Programs
216
Multimedia Application Program 224
Other Program Modules 218
Program Data
220
Processing Unit(s)
202
206
Bus
Video Adapter
244
Output Peripheral Interface
246
Network
248
LAN
251
User Input Interface
242
Data Media Interfaces
226
228
230
232
234
236
238
240
Keyboard
252
WAN
Modem
254
250
Remote Applications
216
Operating System
214
Application Programs
216
Other Modules
218
Program Data
220

Fig. 2

Fig. 4


410
Media Time
308
Matrix Switch
Input 0
Output 0
406
Matrix Switch
Programming Grid
402
404
408
Input/Output
Buffer(s)
Input N
Output N
Project Time
412

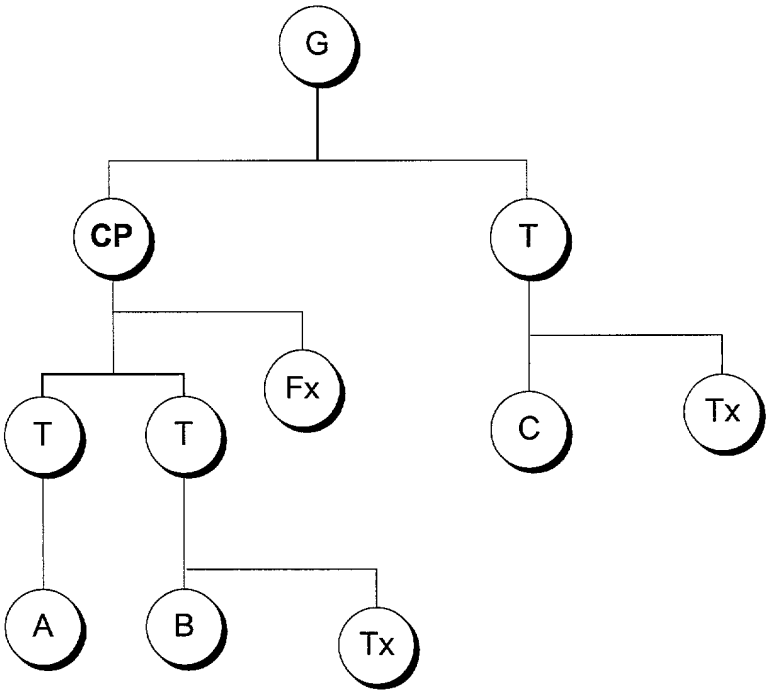

G
CP
T
Fx
T
T
C
Tx
A
B
Tx

Fig. 38a

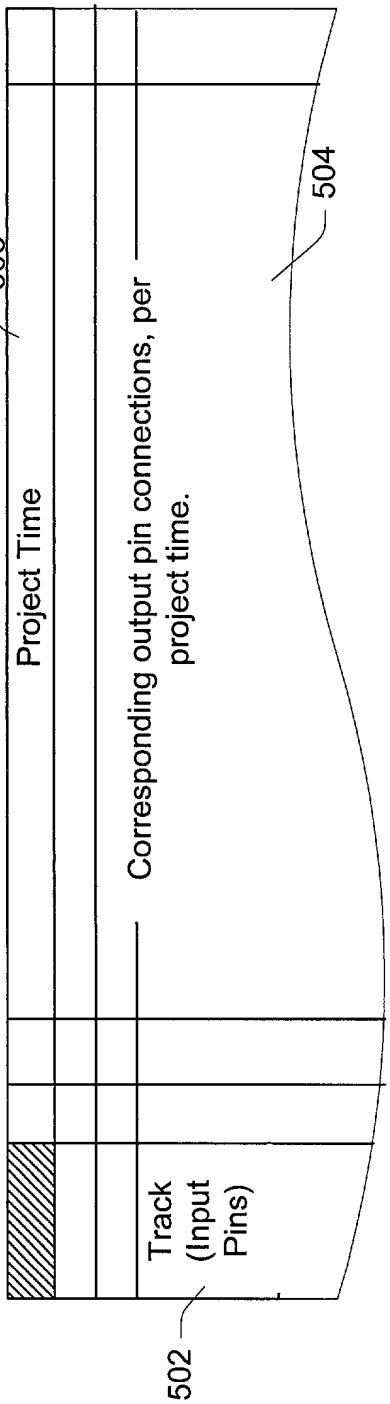

406
Project Time
506
Corresponding output pin connections, per project time.
504
Track (Input Pins)
502

Fig. 5

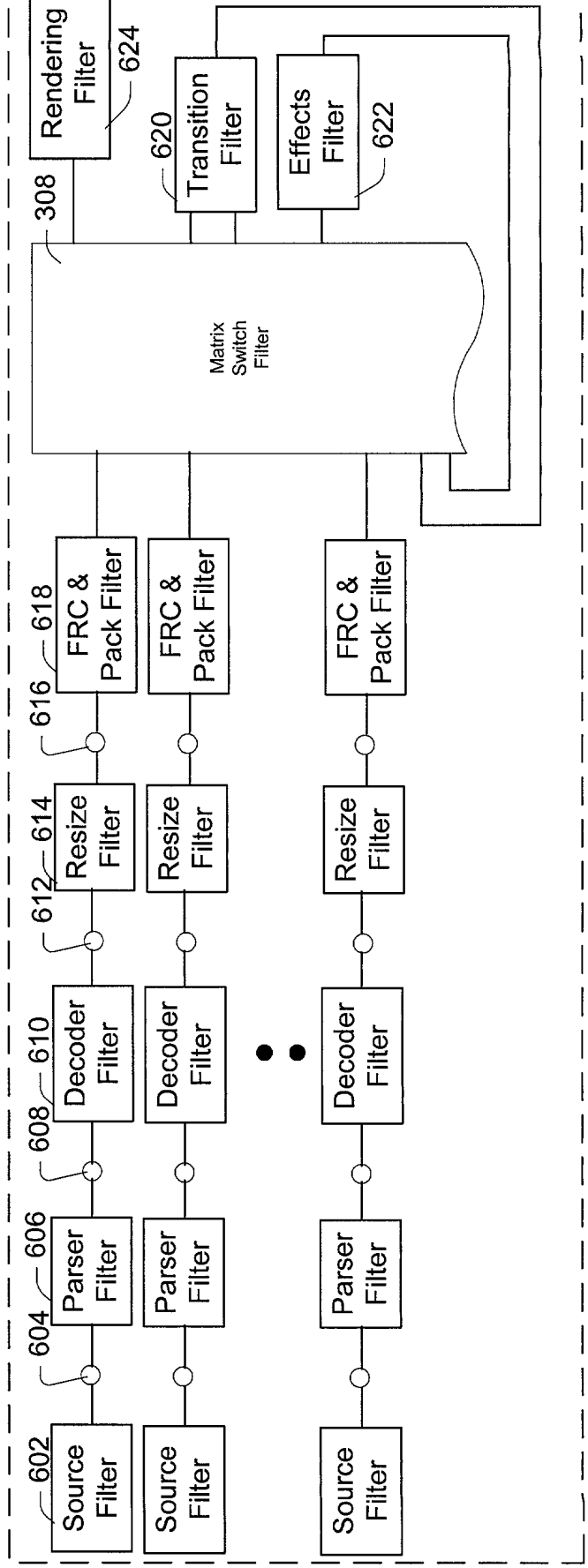

600
Rendering Filter
624
308
Transition Filter
620
Effects Filter
622
Matrix Switch Filter
FRC & Pack Filter
618
Resize Filter
614
Decoder Filter
610
Parser Filter
606
Source Filter
602
616
612
608
604

Fig. 6

Fig. 7

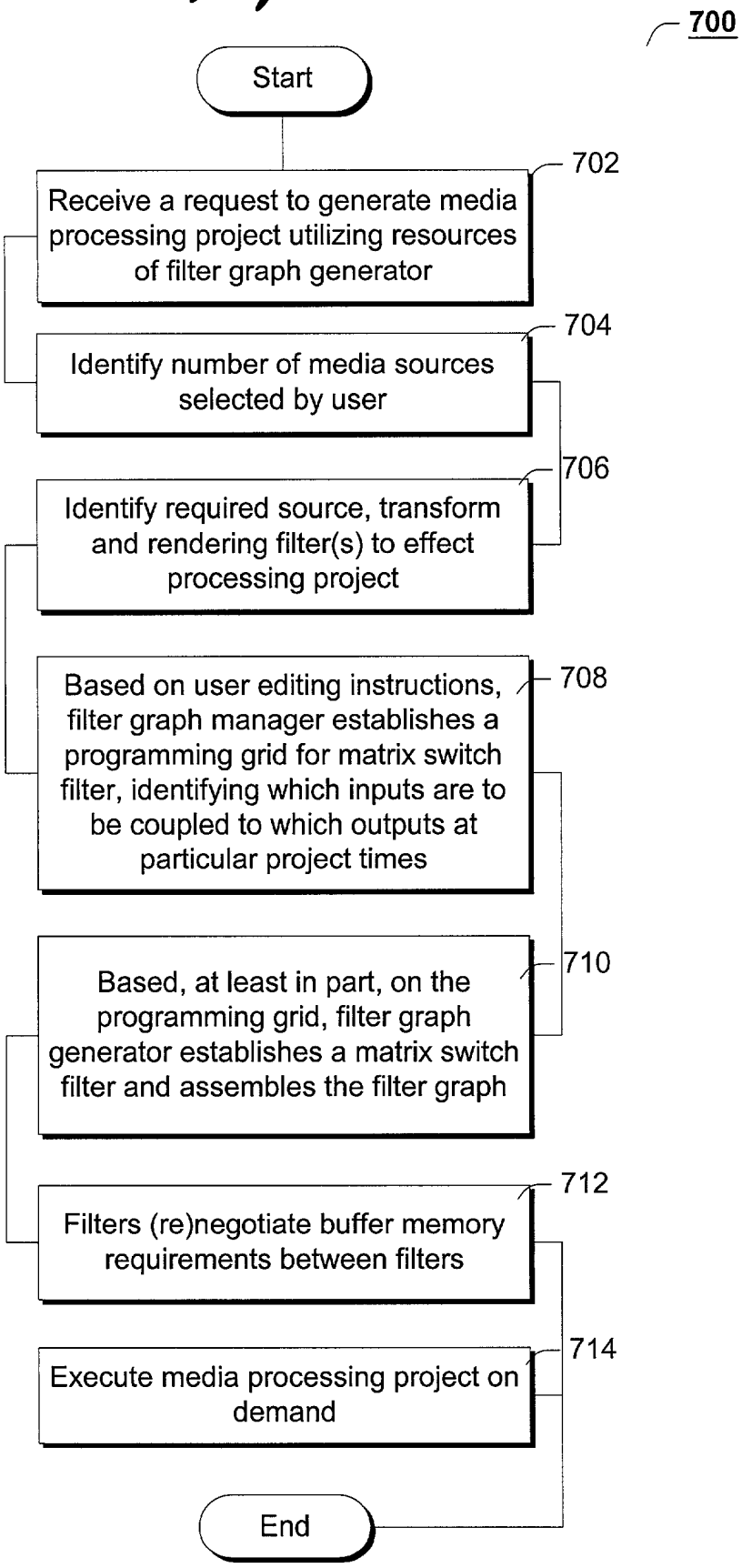
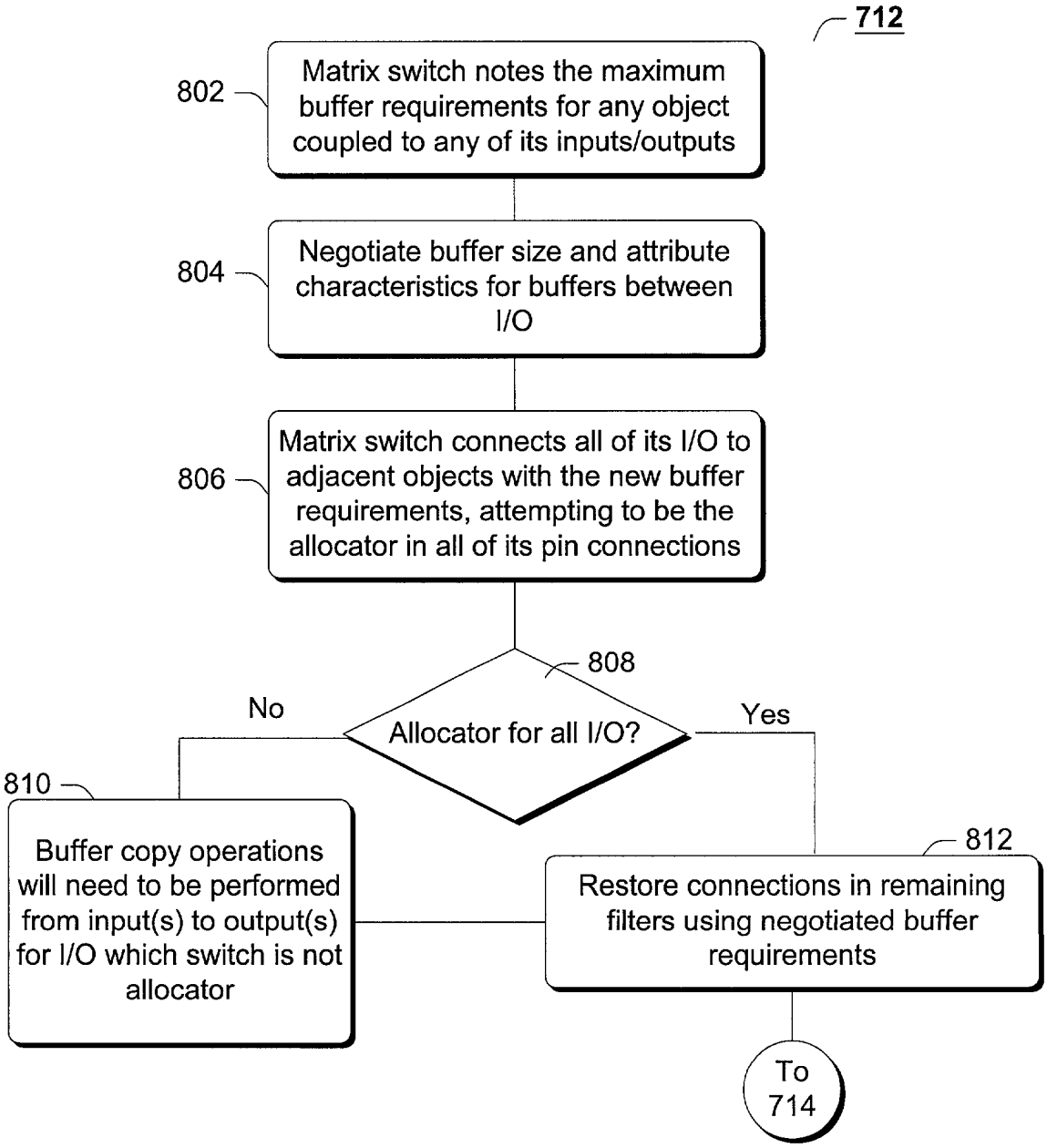

700
Start
702
Receive a request to generate media processing project utilizing resources of filter graph generator
704
Identify number of media sources selected by user
706
Identify required source, transform and rendering filter(s) to effect processing project
708
Based on user editing instructions, filter graph manager establishes a programming grid for matrix switch filter, identifying which inputs are to be coupled to which outputs at particular project times
710
Based, at least in part, on the programming grid, filter graph generator establishes a matrix switch filter and assembles the filter graph
712
Filters (re)negotiate buffer memory requirements between filters
714
Execute media processing project on demand
End

Fig. 8


712
802
Matrix switch notes the maximum buffer requirements for any object coupled to any of its inputs/outputs
804
Negotiate buffer size and attribute characteristics for buffers between I/O
806
Matrix switch connects all of its I/O to adjacent objects with the new buffer requirements, attempting to be the allocator in all of its pin connections
808
Allocator for all I/O?
No
Yes
810
Buffer copy operations will need to be performed from input(s) to output(s) for I/O which switch is not allocator
812
Restore connections in remaining filters using negotiated buffer requirements
To 714

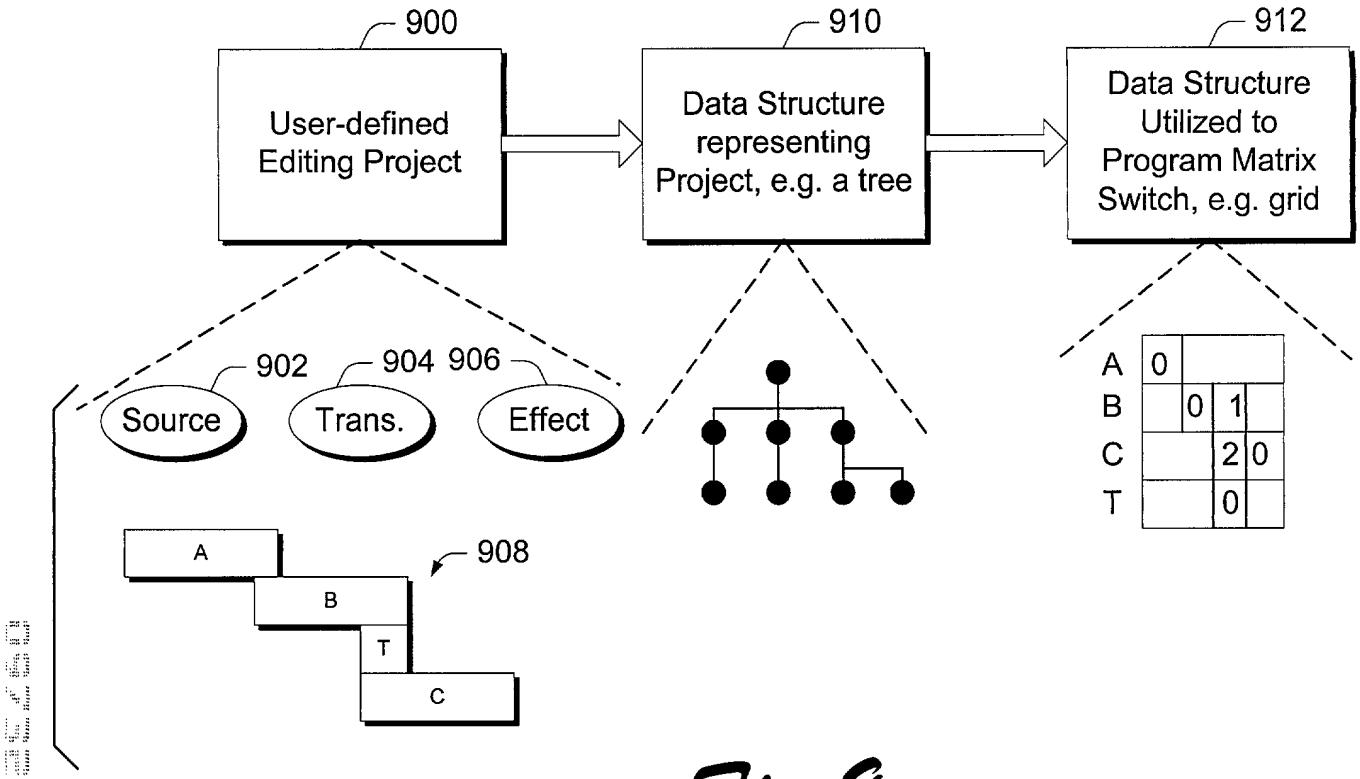

900
User-defined Editing Project
910
Data Structure representing Project, e.g. a tree
912
Data Structure Utilized to Program Matrix Switch, e.g. grid
902
Source
904
Trans.
906
Effect
A
B
T
C
908
A 0
B 0 1
C 2 0
T 0

Fig. 9

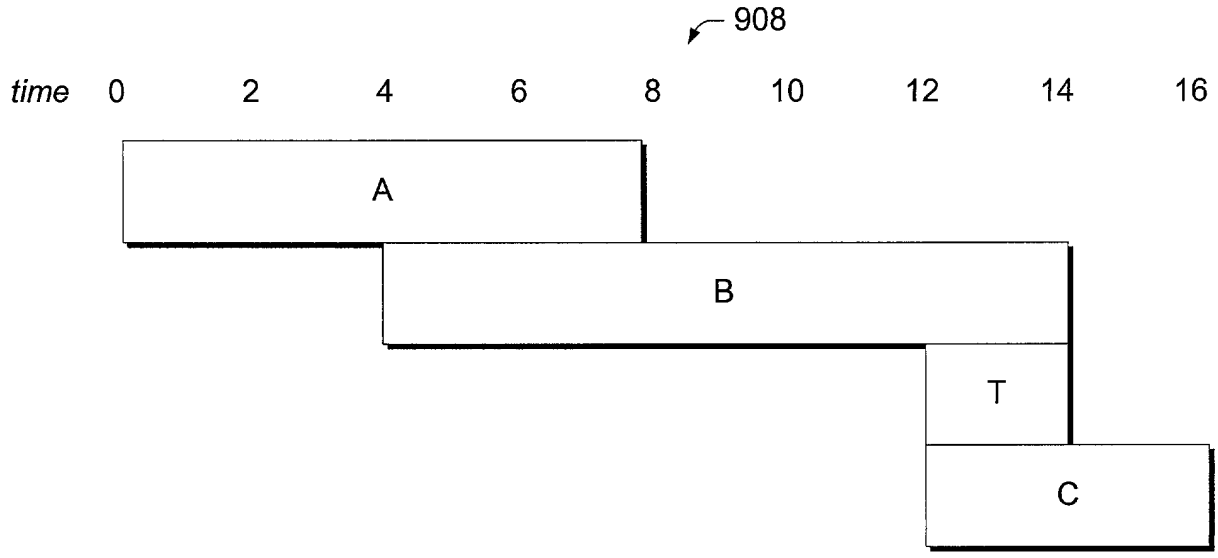

908
time 0 2 4 6 8 10 12 14 16
A
B
T
C

Fig. 10

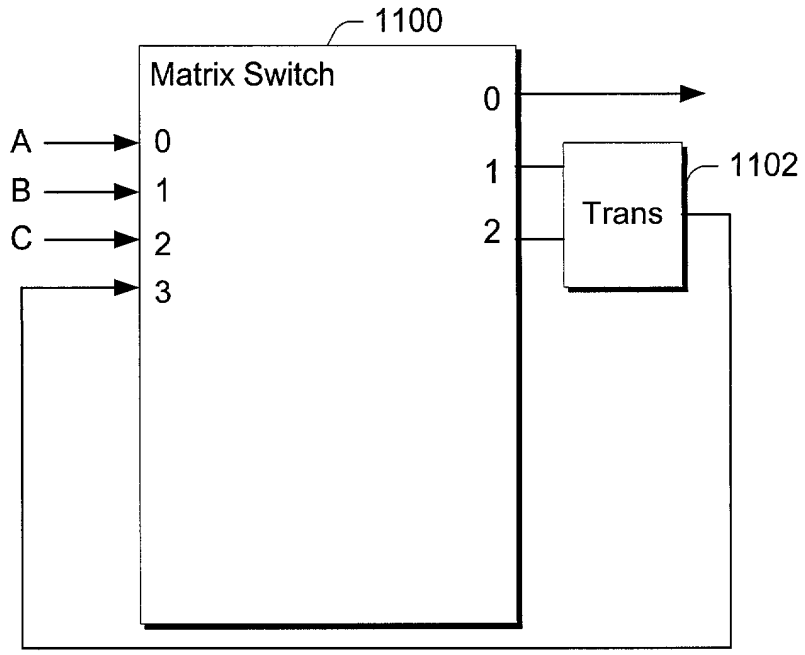

1100
Matrix Switch
A
B
C
0
1
2
3
0
1
2
Trans
1102

Fig. 11

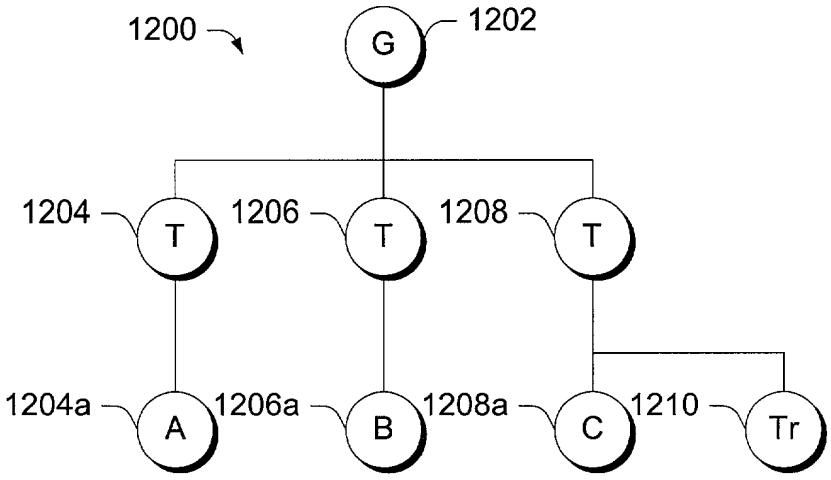

1200
G
1202
1204
T
1206
T
1208
T
1204a
A
1206a
B
1208a
C
1210
Tr

Fig. 12

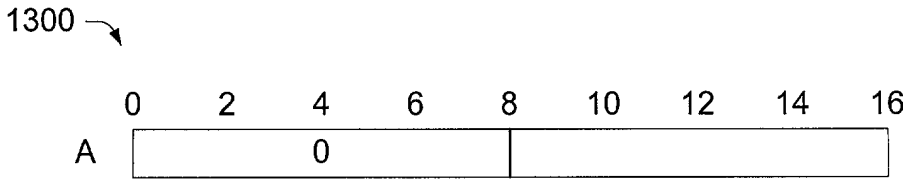


Fig. 13

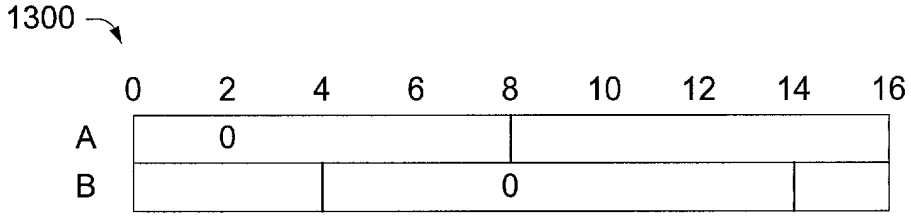


Fig. 14

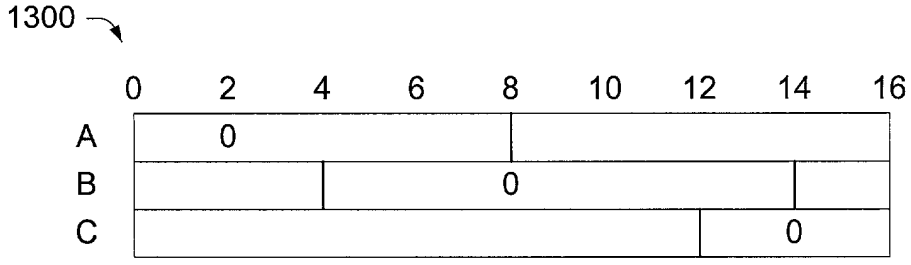


Fig. 15

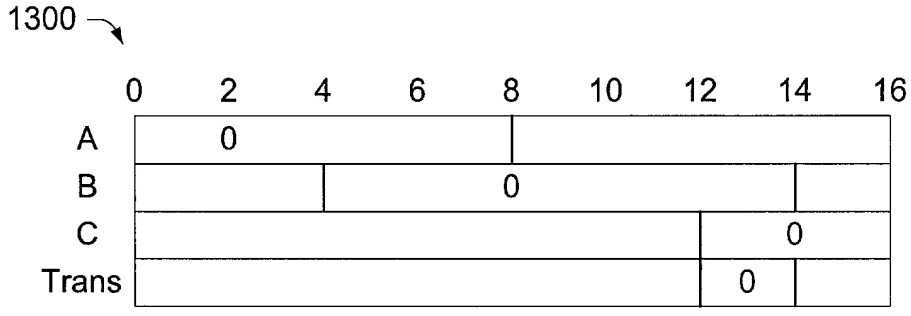


Fig. 16

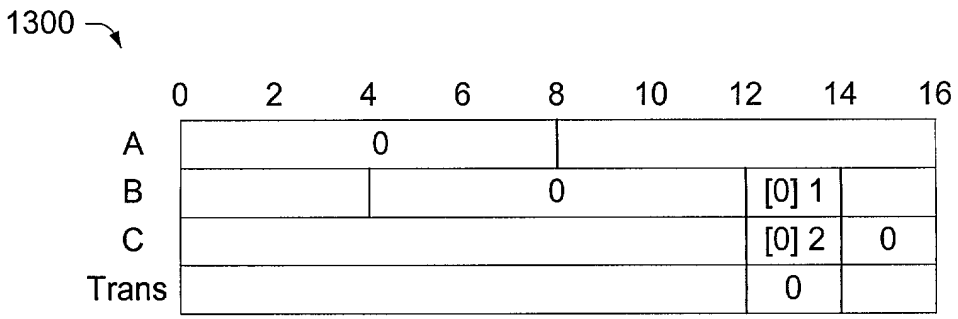

1300
0 2 4 6 8 10 12 14 16
A
B
C
Trans
0
0
[0] 1
[0] 2
0
0

Fig. 17

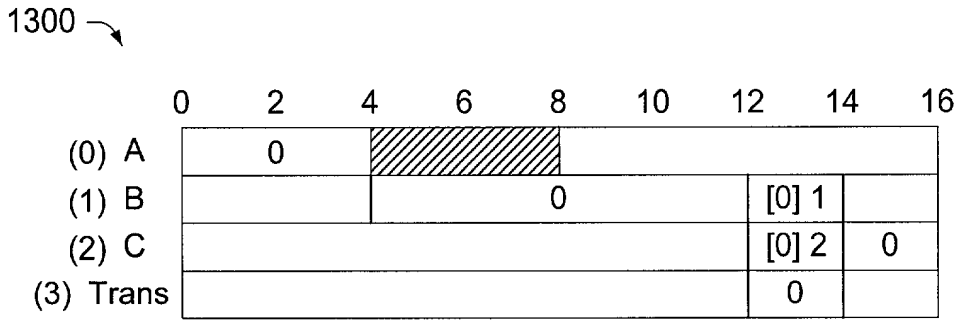

1300
0 2 4 6 8 10 12 14 16
(0) A
(1) B
(2) C
(3) Trans
0
0
[0] 1
[0] 2
0
0

Fig. 18

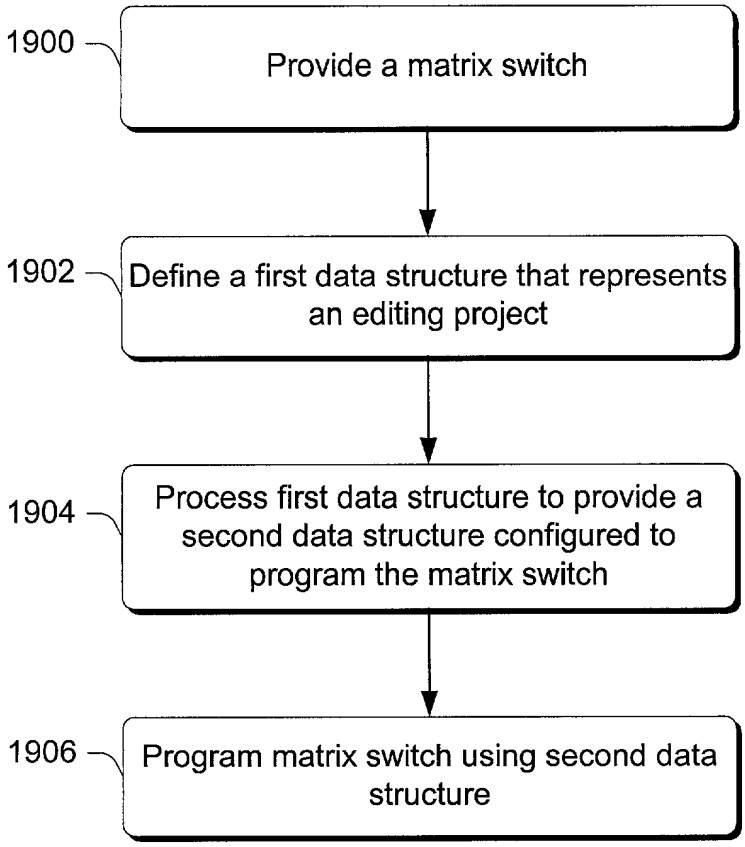

1900
Provide a matrix switch
1902
Define a first data structure that represents an editing project
1904
Process first data structure to provide a second data structure configured to program the matrix switch
1906
Program matrix switch using second data structure

Fig. 19

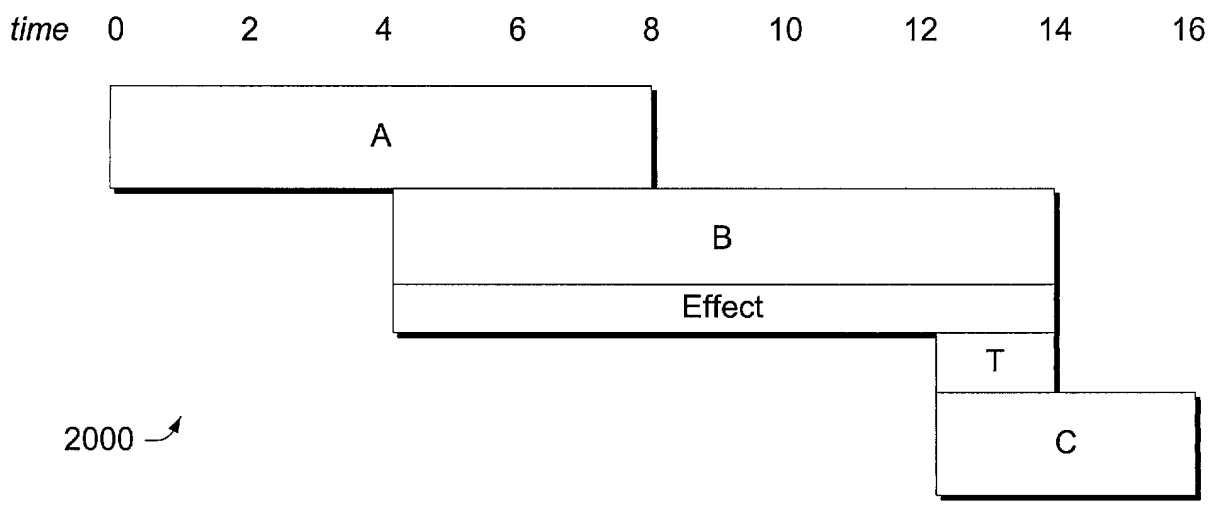


Fig. 20

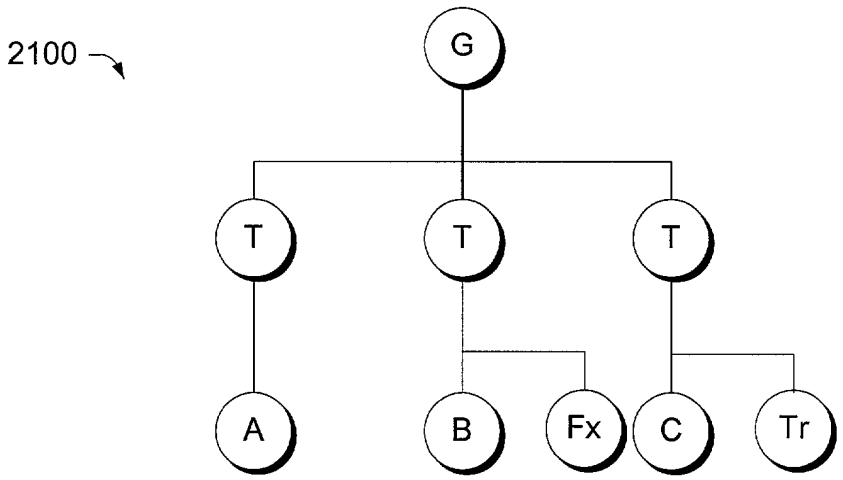


Fig. 21

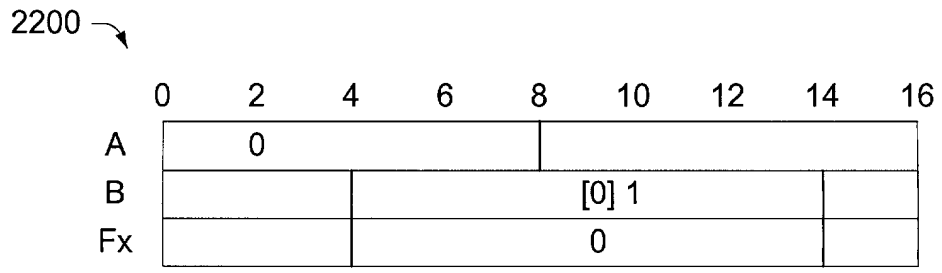


Fig. 22

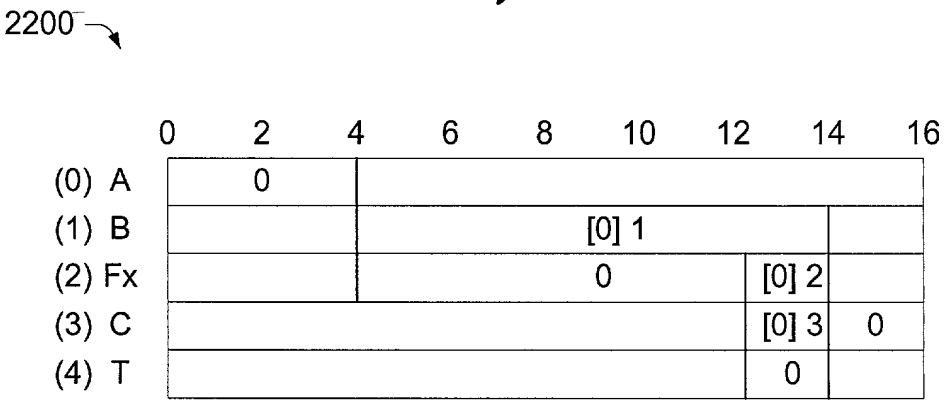


Fig. 23

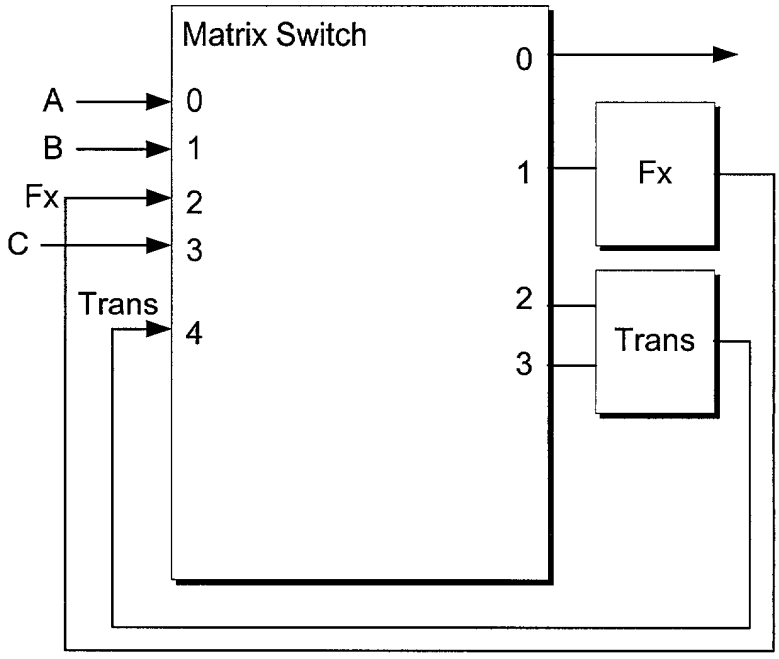


Fig. 24

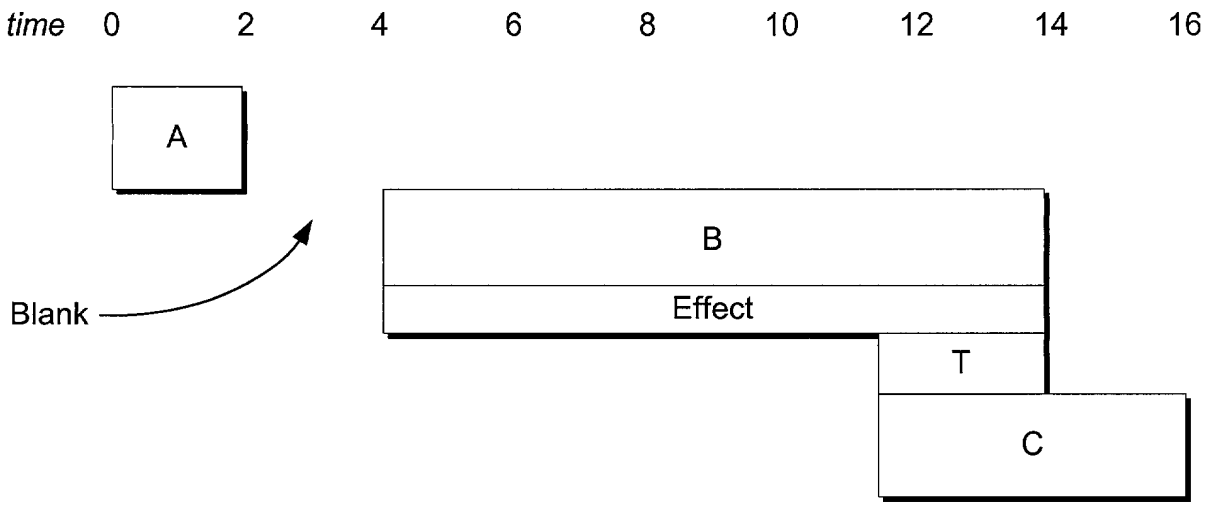

time
0
2
4
6
8
10
12
14
16
A
B
Effect
T
C
Blank

Fig. 25

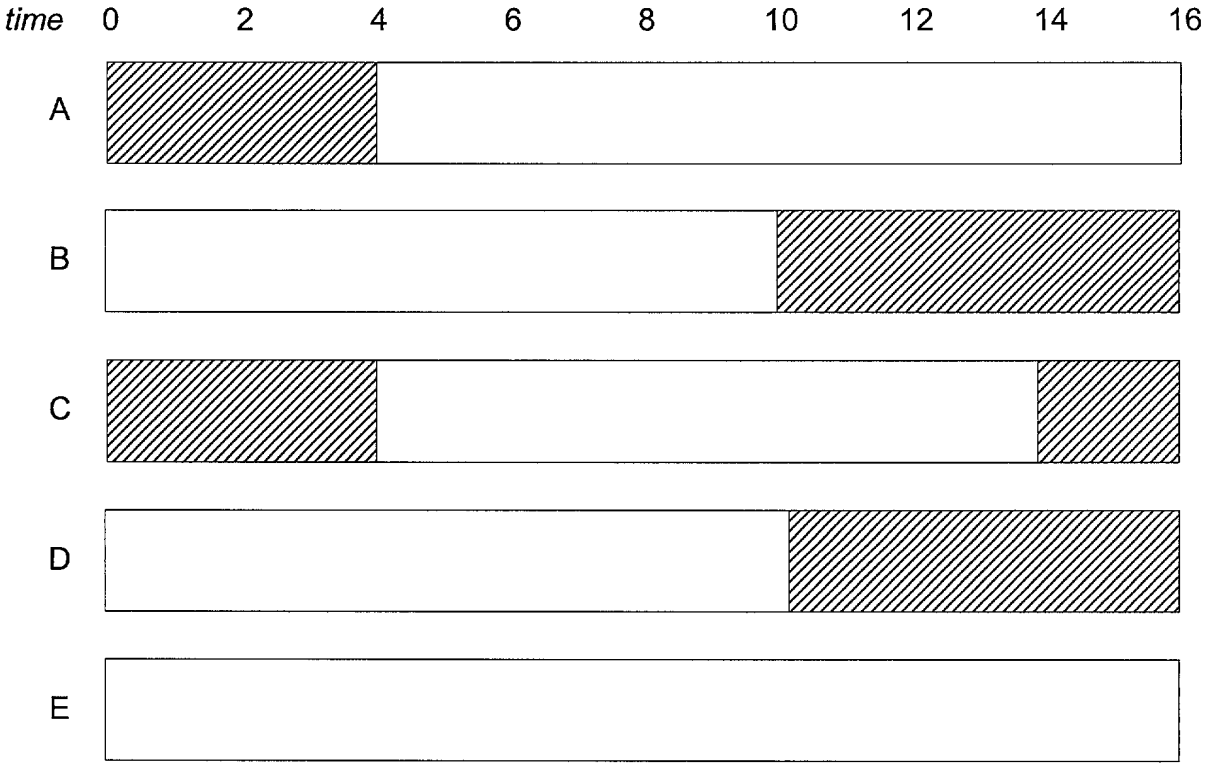

time
0
2
4
6
8
10
12
14
16
A
B
C
D
E

Fig. 26

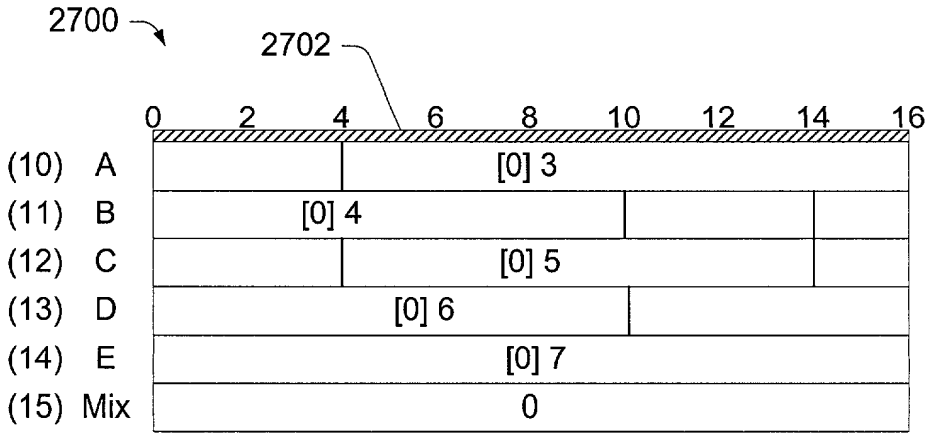

2700
2702
0 2 4 6 8 10 12 14 16
(10) A
(11) B
(12) C
(13) D
(14) E
(15) Mix
[0] 3
[0] 4
[0] 5
[0] 6
[0] 7
0

Fig. 27

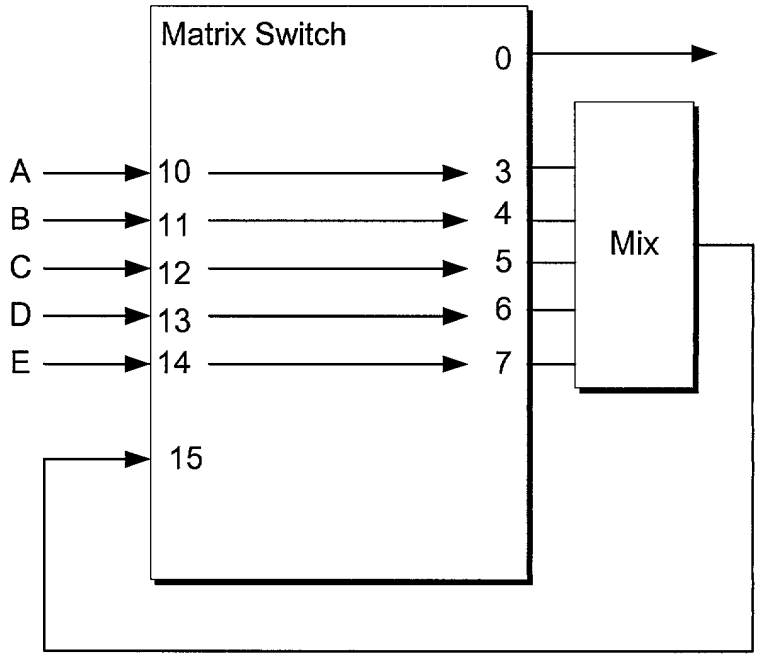

Matrix Switch
A
B
C
D
E
10
11
12
13
14
15
0
3
4
5
6
7
Mix

Fig. 28

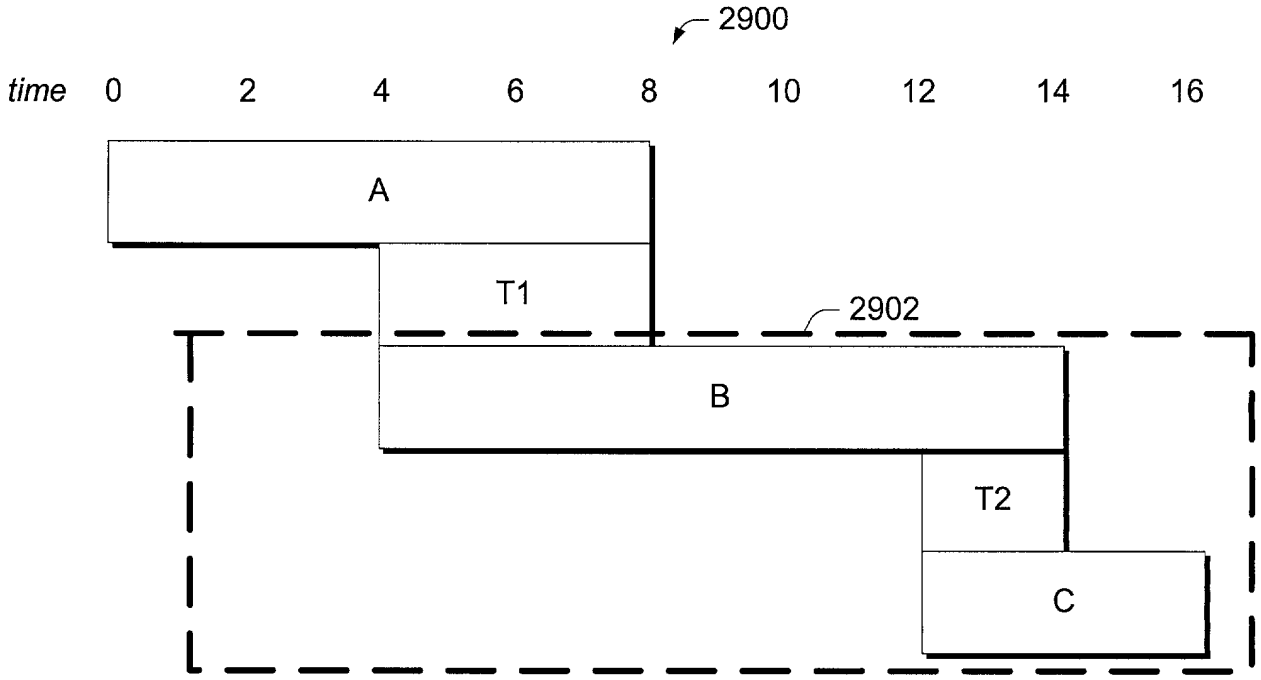

2900
time
0
2
4
6
8
10
12
14
16
A
T1
2902
B
T2
C

Fig. 29

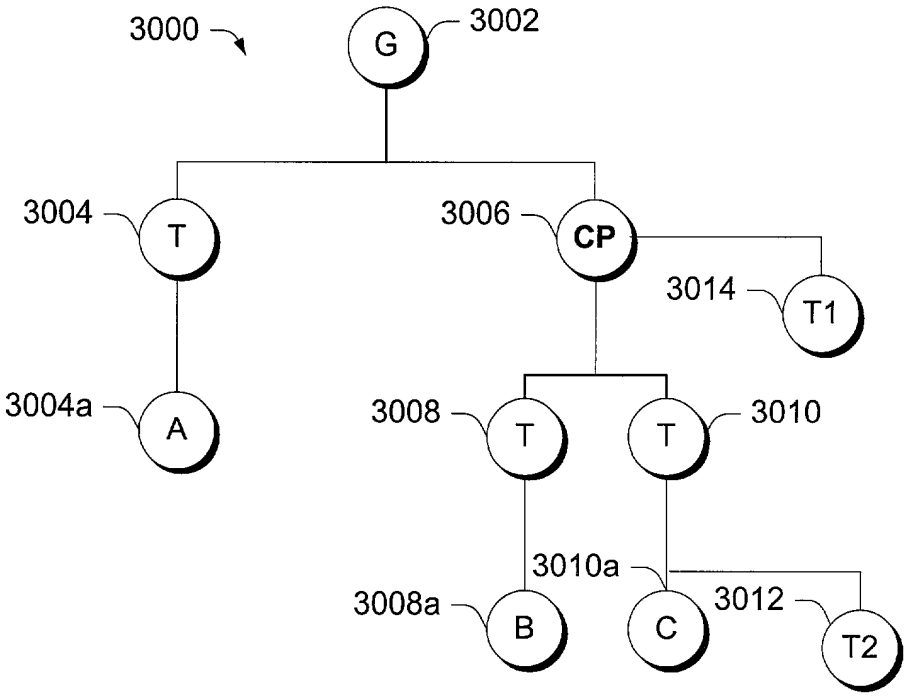

3000
G
3002
3004
T
3006
CP
3014
T1
3004a
A
3008
T
T
3010
3010a
3008a
B
C
3012
T2

Fig. 30

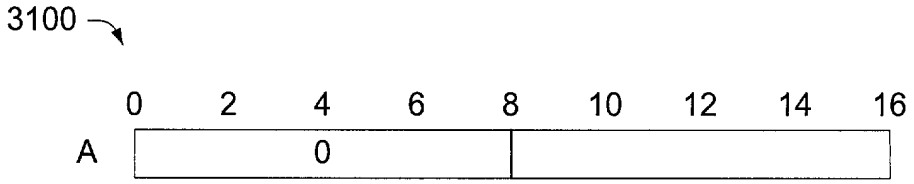


Fig. 31

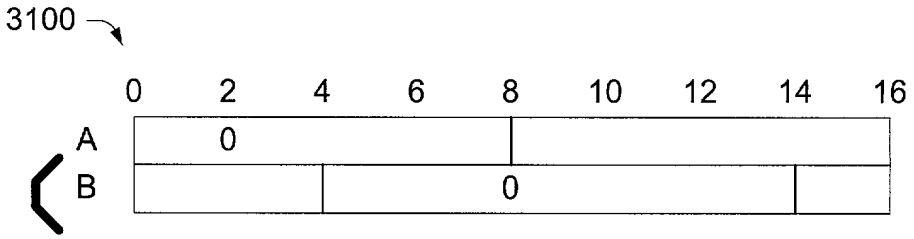


Fig. 32

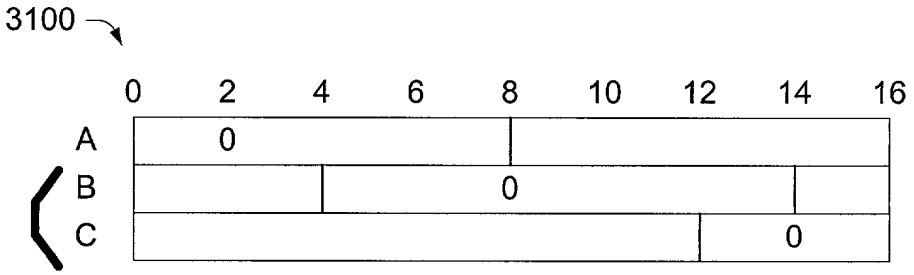


Fig. 33

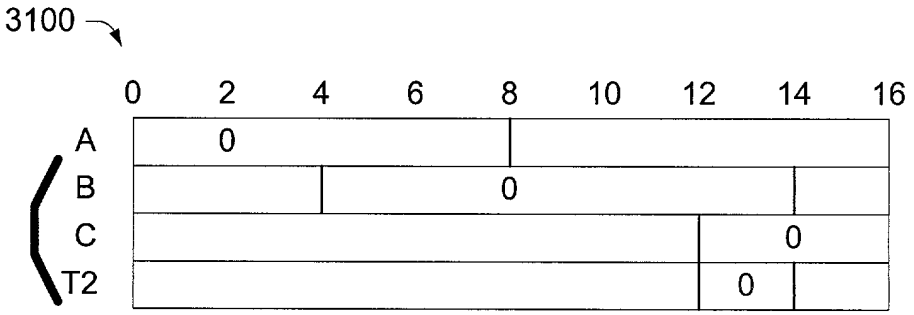


Fig. 34

Patent Application Publication Sep. 7, 2006 Sheet 29 of 29 US 2006/0200778 A1

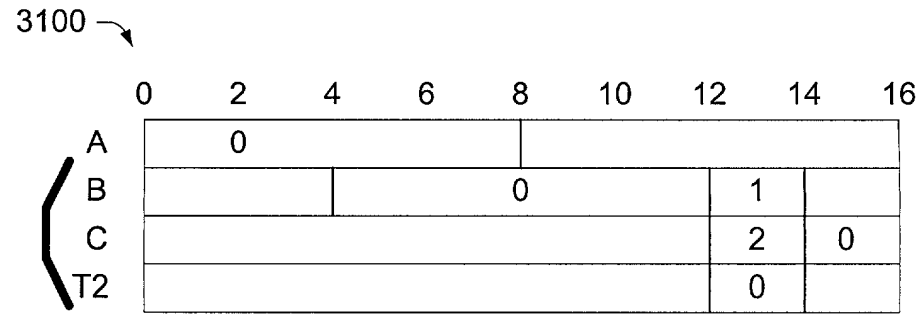


Fig. 35

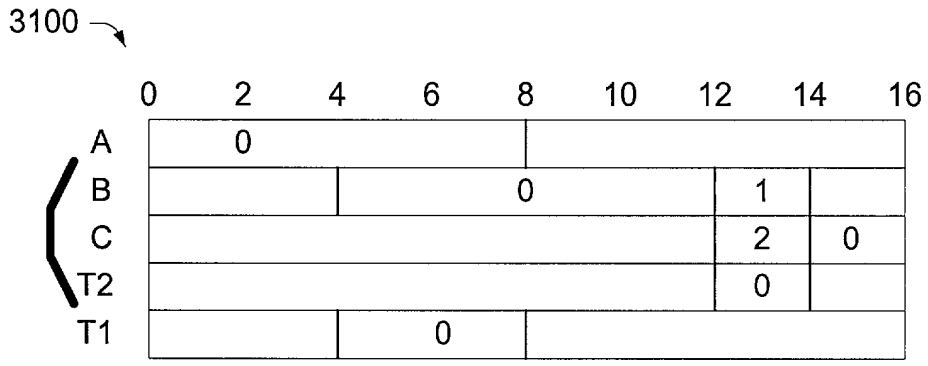


Fig. 36

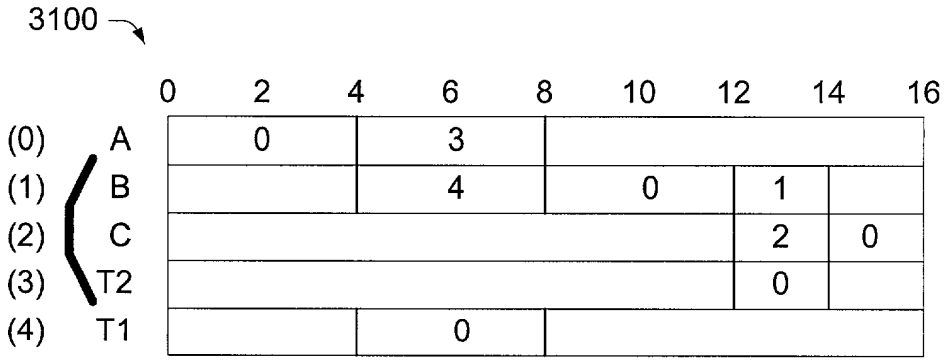


Fig. 37

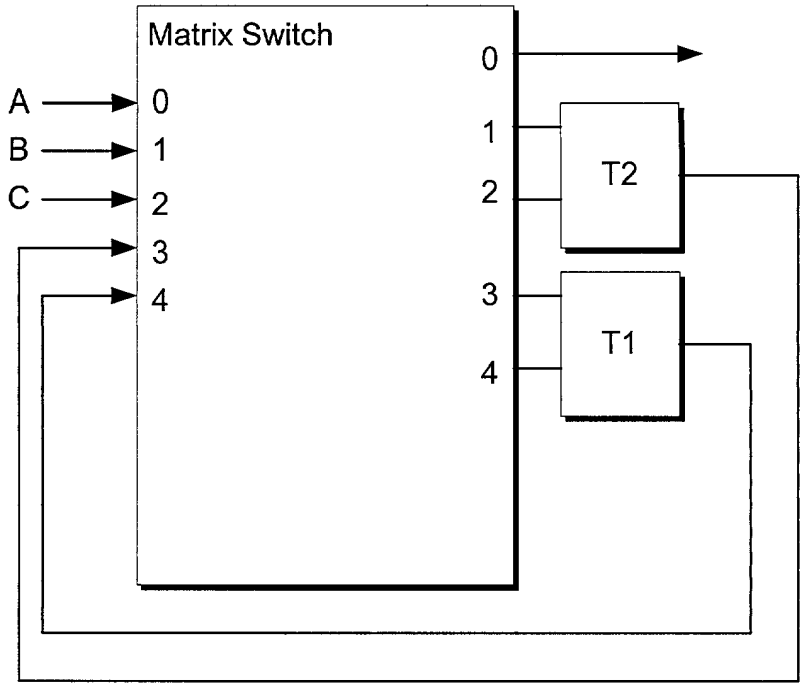

Matrix Switch
A
B
C
0
1
2
3
4
0
1
2
3
4
T2
T1

Fig. 38

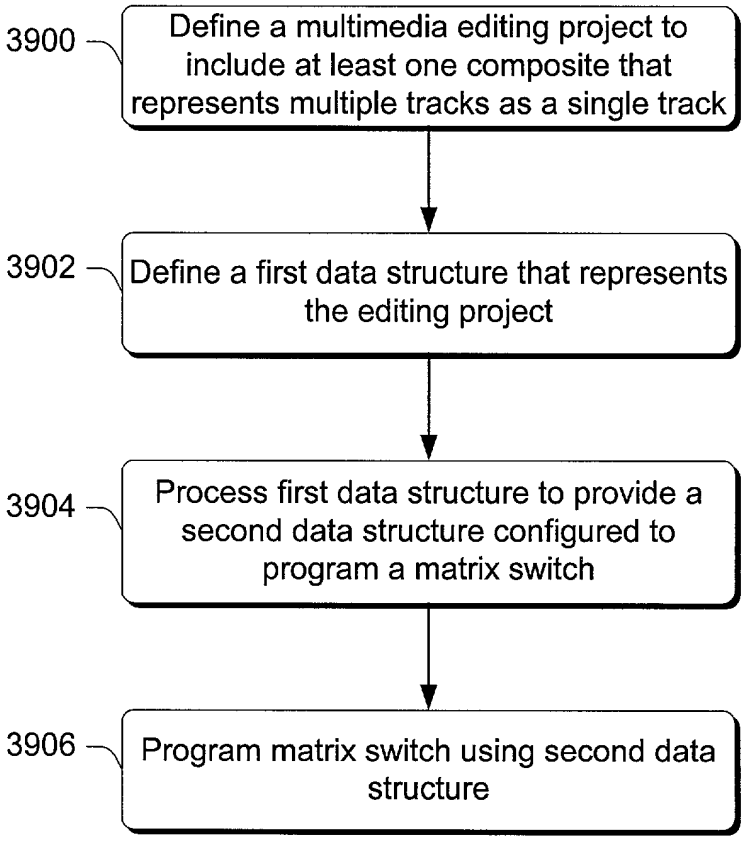

3900
Define a multimedia editing project to include at least one composite that represents multiple tracks as a single track
3902
Define a first data structure that represents the editing project
3904
Process first data structure to provide a second data structure configured to program a matrix switch
3906
Program matrix switch using second data structure

Fig. 39

Fig. 40

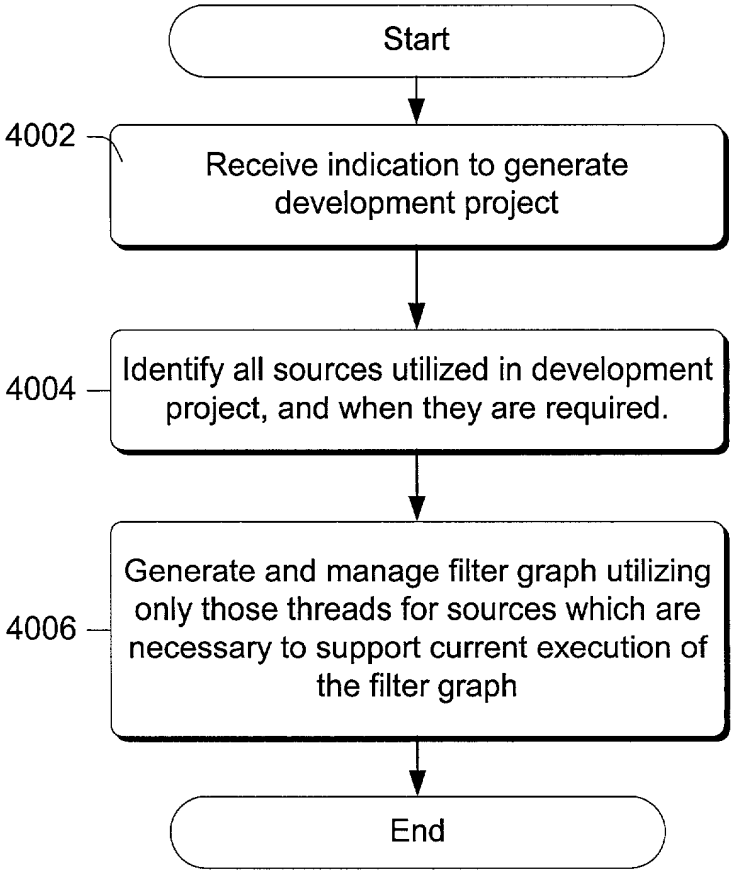

Start
4002
Receive indication to generate development project
4004
Identify all sources utilized in development project, and when they are required.
4006
Generate and manage filter graph utilizing only those threads for sources which are necessary to support current execution of the filter graph
End

Fig. 41

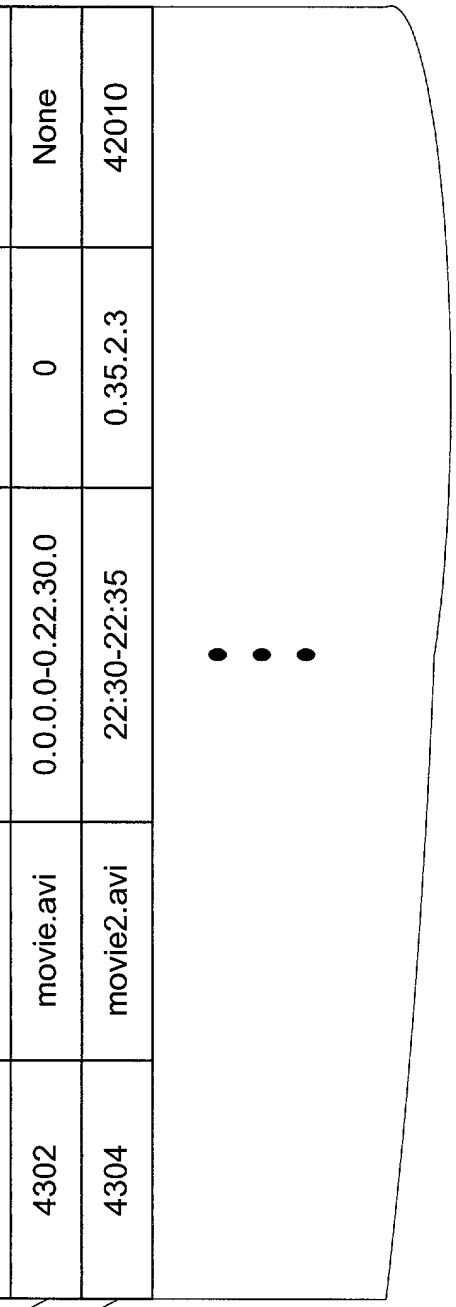
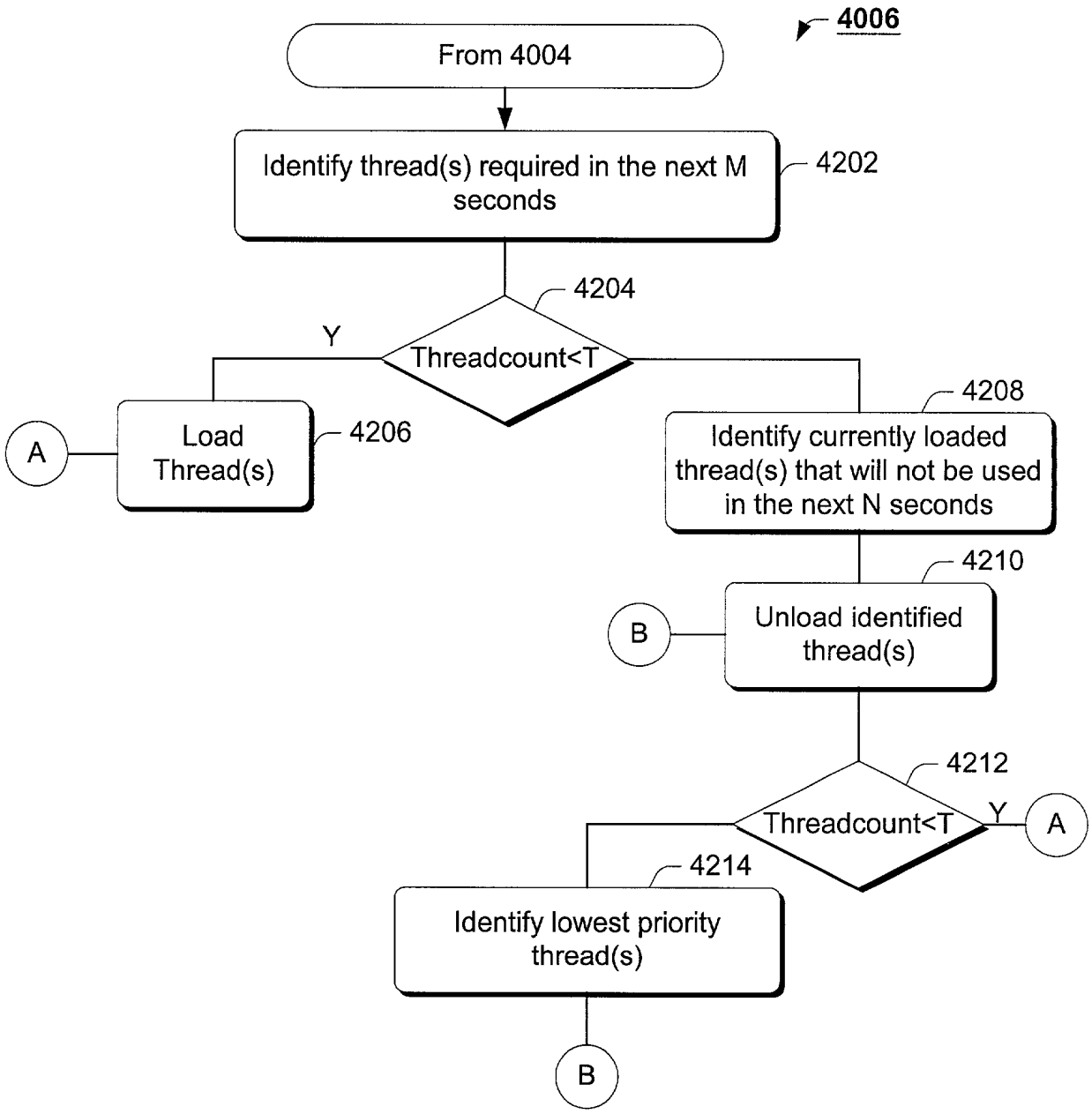

4100
4102
4104
4106
4108
4110
Thread ID
Source ID
Project Time
Source Time
Dependencies
4302
movie.avi
0.0.0.0-0.22.30.0
0
None
4304
movie2.avi
22:30-22:35
0.35.2.3
42010
4112
4114

Fig. 42


4006
From 4004
Identify thread(s) required in the next M seconds
4202
4204
Threadcount<T
Y
A
Load Thread(s)
4206
4208
Identify currently loaded thread(s) that will not be used in the next N seconds
4210
B
Unload identified thread(s)
4212
Threadcount<T
Y
A
4214
Identify lowest priority thread(s)
B

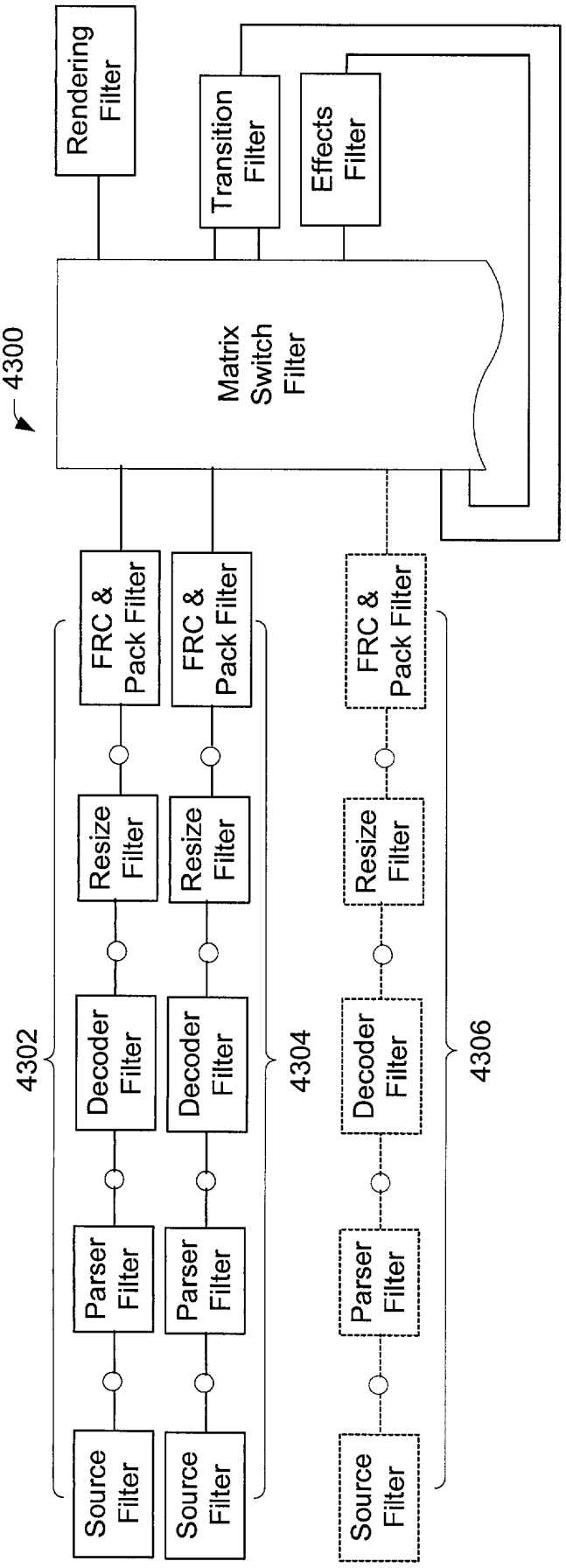

4300
Matrix Switch Filter
Rendering Filter
Transition Filter
Effects Filter
4302
4304
4306
Source Filter
Parser Filter
Decoder Filter
Resize Filter
FRC & Pack Filter

Fig. 43

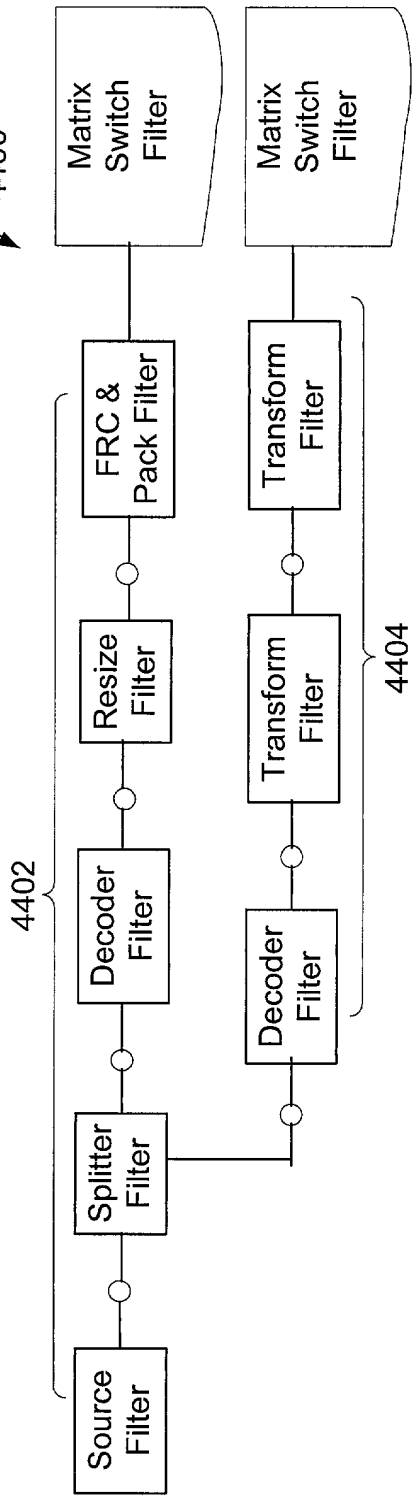

4400
Matrix Switch Filter
Matrix Switch Filter
4402
4404
Source Filter
Splitter Filter
Decoder Filter
Resize Filter
FRC & Pack Filter
Decoder Filter
Transform Filter
Transform Filter

Fig. 44

Fig. 45

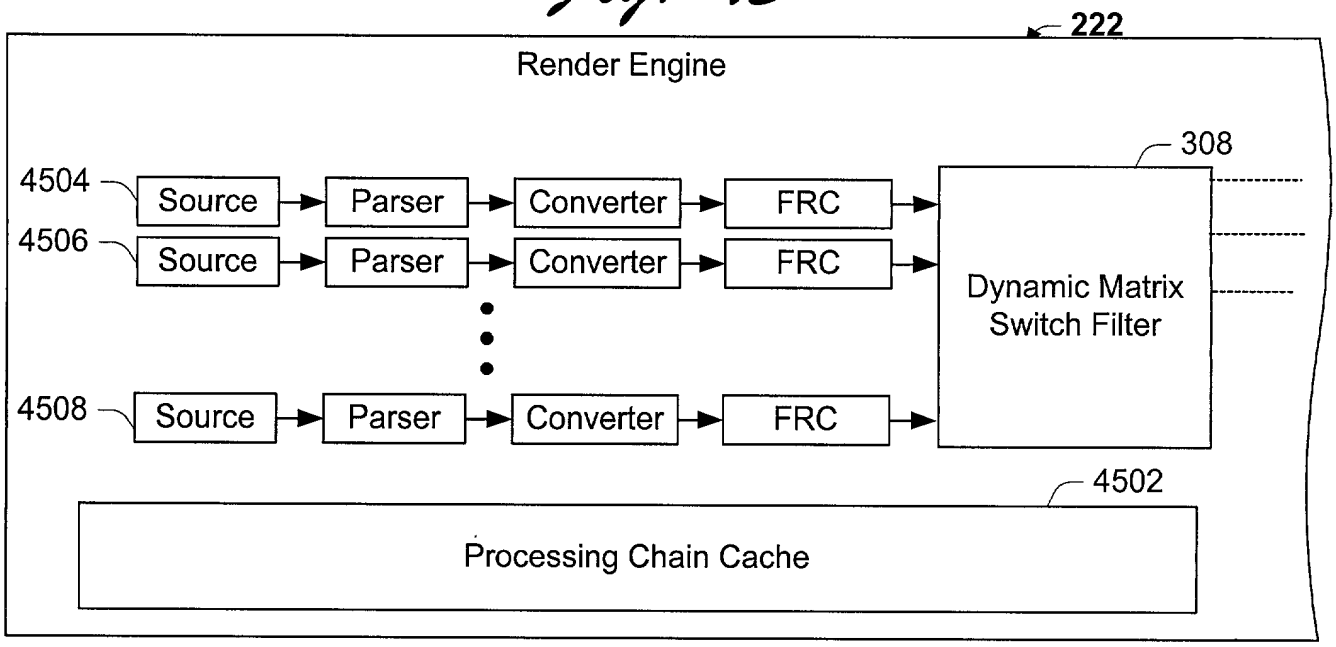

222
Render Engine
308
4504
Source
Parser
Converter
FRC
4506
Source
Parser
Converter
FRC
Dynamic Matrix Switch Filter
4508
Source
Parser
Converter
FRC
4502
Processing Chain Cache

Fig. 46

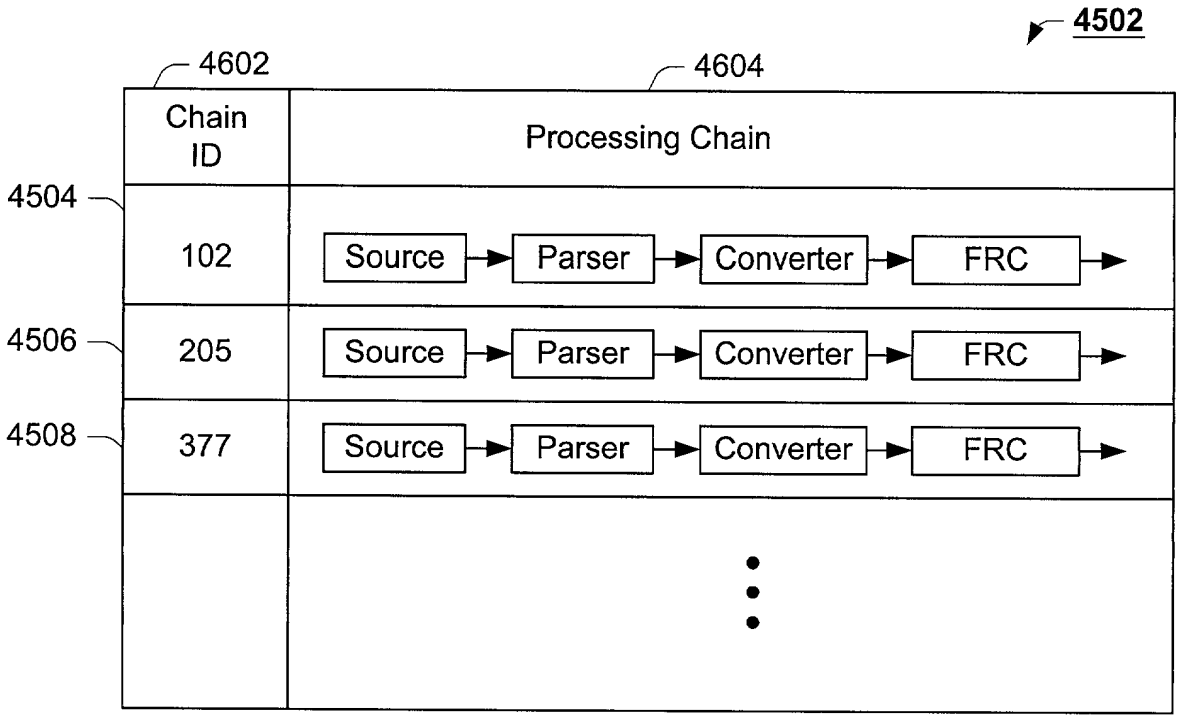

4502
4602
4604
Chain ID
Processing Chain
4504
102
Source
Parser
Converter
FRC
4506
205
Source
Parser
Converter
FRC
4508
377
Source
Parser
Converter
FRC

Fig. 47

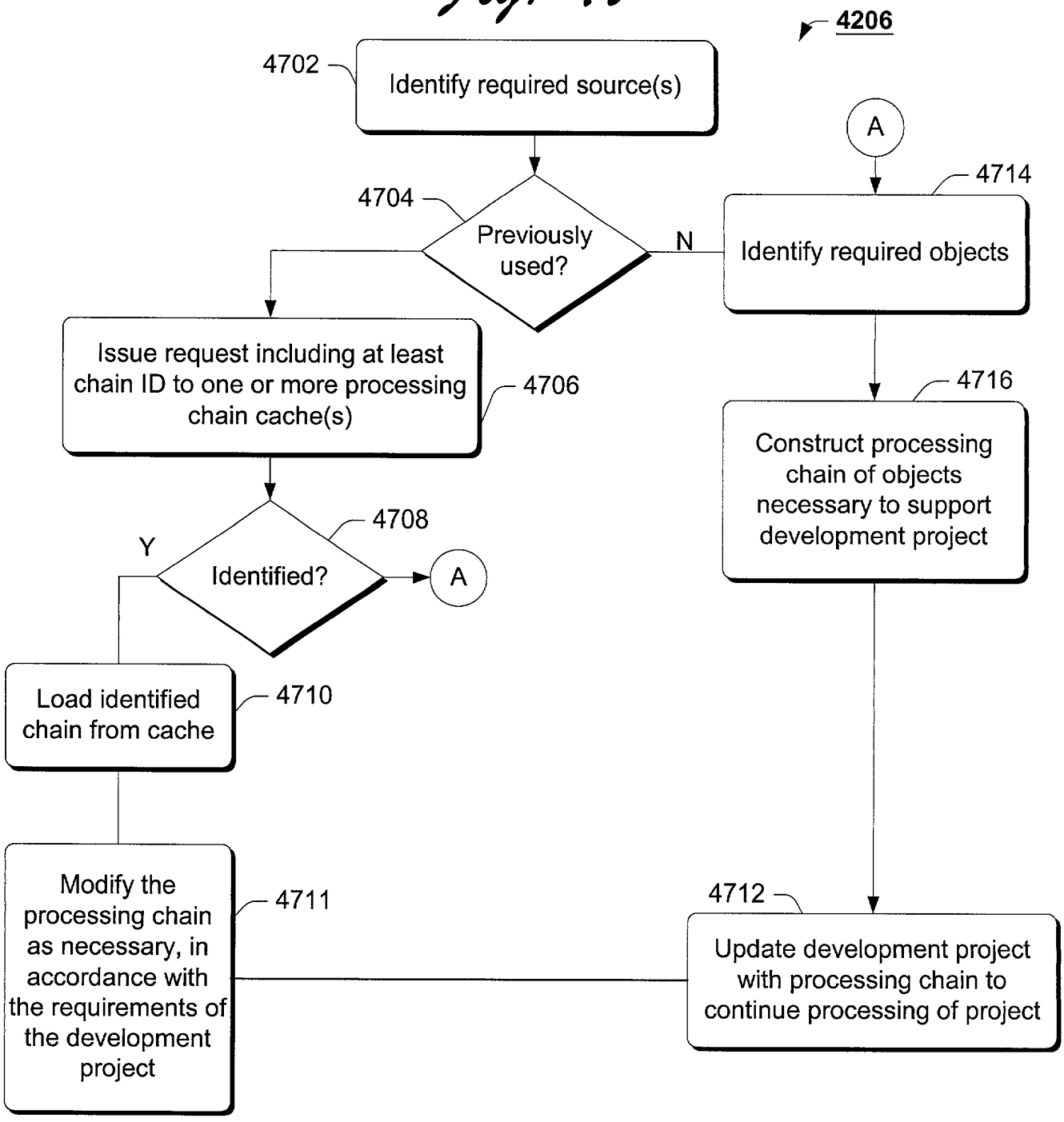
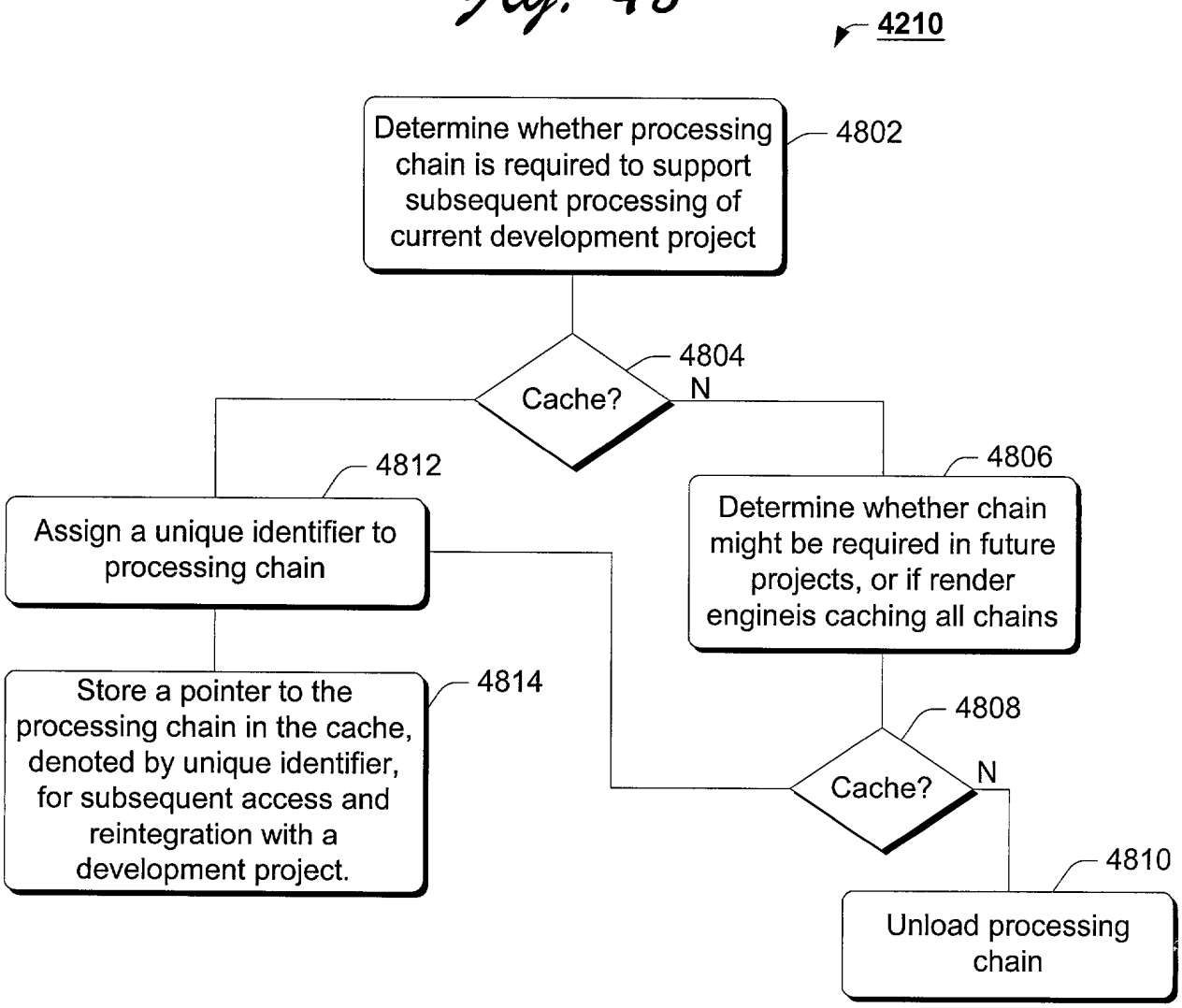

4206
4702
Identify required source(s)
4704
Previously used?
N
Y
A
4714
Identify required objects
Issue request including at least chain ID to one or more processing chain cache(s)
4706
4716
Construct processing chain of objects necessary to support development project
4708
Identified?
A
Load identified chain from cache
4710
Modify the processing chain as necessary, in accordance with the requirements of the development project
4711
4712
Update development project with processing chain to continue processing of project

Fig. 48


4210
Determine whether processing chain is required to support subsequent processing of current development project
4802
Cache?
4804
N
4812
Assign a unique identifier to processing chain
4806
Determine whether chain might be required in future projects, or if render engineis caching all chains
Store a pointer to the processing chain in the cache, denoted by unique identifier, for subsequent access and reintegration with a development project.
4814
4808
Cache?
N
4810
Unload processing chain